**JAVA and JDBC code**

**Submitted By: BHAVIN DESAI, SWETHA SUNDARARAJAN**

1. **Driver.java**

**package** jdbc\_trial;

**import** java.sql.SQLException;

**import** java.util.InputMismatchException;

**import** java.util.Scanner;

/\*\*

\* **@author** Bhavin

\* This is the main driver code where you will get all the user input

\* and feed all parameters to the class and call approriate methods.

\* Throw exceptions appropriately

\*/

**public** **class** Driver {

**private** **static** Scanner *reader*;

/\*\*

\* **@param** args

\* Main function

\*/

**public** **static** **void** main(String[] args) {

**while**(**true**){

System.***out***.println("\n--> Enter [1..10] for running your desired query");

System.***out***.println(" 2. Display 3. Add Student 4. Show Details 5. Show Prerequisites");

System.***out***.println(" 6. Show Student Enrollment 7. Enroll Student 8. Drop Class 9. Delete Student\n");

System.***out***.print("Input Number :");

*reader* = **new** Scanner(System.***in***);

**int** query\_no = -1;

**try**{

query\_no = *reader*.nextInt();

}**catch**(InputMismatchException e){e.getMessage();}

// validate your input

**if**(query\_no <1 && query\_no>10){

System.***out***.println("\nError: Invalid selection\n");

System.*exit*(0);

}

**switch**(query\_no){

**case** 1: // Query one: The sequence

System.***out***.println("The sequence is generated through script.");

**break**;

**case** 2: // Query 2

System.***out***.print("Enter table name:");

Scanner tableinput = **new** Scanner(System.***in***);

String tablename = tableinput.next();

Query2\_display data = **new** Query2\_display(tablename);

**try** {

data.show\_data();

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}

**break**;

**case** 3: // Query 3 : add student

System.***out***.print("\nEnter Student Details (Comma Seperated):");

Scanner stud = **new** Scanner(System.***in***);

String stud\_details = stud.next();

add\_student as = **new** add\_student(stud\_details);

**try** {

as.add\_stud();

} **catch** (SQLException e1) {

System.***out***.println(e1.getMessage());

}

**break**;

**case** 4: // Query 4

System.***out***.print("\nEnter SID: ");

Scanner sid = **new** Scanner(System.***in***);

String stud\_id = sid.next();

// pass parameters and call the method

four\_showdetails fsd = **new** four\_showdetails(stud\_id);

**try** {

fsd.show\_details();

} **catch** (SQLException e1) {

System.***out***.println(e1.getMessage());

}

**break**;

**case** 5: // Query 5

System.***out***.print("\nEnter Dept Code: ");

Scanner five= **new** Scanner(System.***in***);

String deptcode= five.next();

System.***out***.print("\nEnter Course Number: ");

String courseno = five.next();

// pass parameters and call the method

Fifth f = **new** Fifth(deptcode, courseno);

**try** {

f.execute\_fifth();

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}

**break**;

**case** 6: //Query 6

System.***out***.print("\nEnter CLASSID: ");

Scanner cid = **new** Scanner(System.***in***);

String classid = cid.next();

// pass parameters and call the method

sixth six = **new** sixth(classid);

**try** {

six.show\_studinclass();

} **catch** (SQLException e1) {

System.***out***.println(e1.getMessage());

}

**break**;

**case** 7: // Query 7: enroll students

// take inputs

System.***out***.print("\nEnter SID: ");

Scanner seven= **new** Scanner(System.***in***);

String sid7= seven.next();

System.***out***.print("\nEnter CLASSID: ");

String classid7 = seven.next();

// pass parameters and call the method

seventh svn = **new** seventh(sid7, classid7);

**try** {

svn.enroll\_student();

} **catch** (SQLException e2) {

System.***out***.println(e2.getMessage());

}

**break**;

**case** 8: // Query 8

System.***out***.print("\nEnter SID: ");

Scanner eight= **new** Scanner(System.***in***);

String sid8= eight.next();

System.***out***.print("\nEnter CLASSID: ");

String classid8 = eight.next();

// pass parameters and call the method

Eighth ei = **new** Eighth(sid8, classid8);

**try** {

ei.drop\_class();

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}

**break**;

**case** 9: // Query 9

System.***out***.print("\nEnter SID to delete: ");

Scanner ssid = **new** Scanner(System.***in***);

String sstud\_id = ssid.next();

// pass parameters and call the method

Ninth\_del\_stud nds = **new** Ninth\_del\_stud(sstud\_id);

**try** {

nds.delete\_student();

} **catch** (SQLException e1) {

System.***out***.println(e1.getMessage());

}

**break**;

**case** 10: System.***out***.println("The triggers are generated through script.");

**break**;

**default**: System.***out***.println("\nWRONG INPUT. Please check again.\n");

**break**;

} //end switch

}//end while

}

}

1. **Query2\_display.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** oracle.jdbc.OracleTypes;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This is the 2nd query which will display all the records of the selected table

\*/

**public** **class** Query2\_display {

String input;

Query2\_display(String tablename){

**this**.input=tablename;

}

**public** **void** show\_data() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs = **null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

cs = **null**;

//Prepare to call stored procedure based on input

**switch**(input){

**case** "students":

cs = conn.prepareCall("{ call display.show\_students(?)}");

**break**;

**case** "courses":

cs = conn.prepareCall("{ call display.show\_courses(?)}");

**break**;

**case** "prerequisites":

cs = conn.prepareCall("{ call display.show\_prerequisites(?)}");

**break**;

**case** "classes":

cs = conn.prepareCall("{ call display.show\_classes(?)}");

**break**;

**case** "enrollments":

cs = conn.prepareCall("{ call display.show\_enrollments(?)}");

**break**;

**case** "logs":

cs = conn.prepareCall("{ call display.show\_logs(?)}");

**break**;

**default**:

System.***out***.println("Invalid selection of a table");

}

// register the out parameter (the first parameter)

cs.registerOutParameter(1, OracleTypes.***CURSOR***);

// execute and retrieve the result set

cs.executeUpdate();

ResultSet rs = (ResultSet)cs.getObject(1);

// print the results according to the column index of the selected table

**while** (rs.next()) {

**switch**(input){

**case** "students" :

System.***out***.println(rs.getString(1) + "\t" + rs.getString(2) + "\t" + rs.getString(3) +"\t"+ rs.getString(4) + "\t" + rs.getDouble(5) + "\t" + rs.getString(6));

**break**;

**case** "courses" :

System.***out***.println(rs.getString(1) + "\t" + rs.getInt(2) + "\t" + rs.getString(3));

**break**;

**case** "prerequisites" :

System.***out***.println(rs.getString(1) + "\t" + rs.getInt(2) + "\t" + rs.getString(3) +"\t"+ rs.getInt(4));

**break**;

**case** "classes" :

System.***out***.println(rs.getString(1) + "\t" + rs.getString(2) + "\t" + rs.getInt(3)+ "\t" + rs.getInt(4) + "\t" + rs.getInt(5) + "\t" + rs.getString(6)

+ "\t" + rs.getInt(7) + "\t" + rs.getInt(8));

**break**;

**case** "enrollments" :

System.***out***.println(rs.getString(1) + "\t" + rs.getString(2) + "\t" + rs.getString(3));

**break**;

**case** "logs" :

System.***out***.println(rs.getInt(1) + "\t" + rs.getString(2) + "\t" + rs.getTime(3) + "\t" +rs.getString(4) + "\t" + rs.getString(5) + "\t" + rs.getString(6));

**break**;

**default**:

System.***out***.println("Invalid number of columns in a table");

}

} //end while

//close the result set, statement, and the connection

cs.close();

conn.close();

System.***out***.println("\nRecords of '"+input+"' table displayed successfully\n");

}

**catch** (SQLException ex) { System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// clode the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **add\_student.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.SQLException;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class helps you to add students based on inputs provided

\* from the driver code through user input

\*/

**public** **class** add\_student {

String stud\_details;

/\*\*

\* **@param** details

\*/

**public** add\_student(String details){

**this**.stud\_details=details;

}

/\*\*

\* **@throws** SQLException

\*/

**public** **void** add\_stud() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs = **null**;

**try**

{

String data[] = stud\_details.split(",");

**if**(data.length!=6){

System.***out***.println("Invalid number of arguments");

}

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{call add\_stud(?,?,?,?,?,?) }");

//set all the in parameters

cs.setString(1, data[0]); //sid

cs.setString(2, data[1]); //firstname

cs.setString(3, data[2]); //lastname

cs.setString(4, data[3]); //status

cs.setDouble(5, Double.*parseDouble*(data[4])); //gpa

cs.setString(6, data[5]); //email

// execute and retrieve the result set

cs.executeUpdate();

System.***out***.println("\n1 Student row inserted successfully");

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) { System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **four\_showdetails.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** oracle.jdbc.OracleTypes;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class is for showing details based on sid

\*/

**public** **class** four\_showdetails {

String sid;

/\*\*

\* **@param** insid

\*/

**public** four\_showdetails(String insid){

**this**.sid=insid;

}

**public** **void** show\_details() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs =**null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{ call show\_details(?,?)}");

//set all the in parameters CHANGE IN HERE

cs.setString(1, sid); //sid

//register the out parameter (the first parameter)

cs.registerOutParameter(2, OracleTypes.***CURSOR***);

// execute and retrieve the result set

cs.executeUpdate();

ResultSet rs = (ResultSet)cs.getObject(2);

// print the results

**while** (rs.next()) {

System.***out***.println(rs.getString(1) + "\t" + rs.getString(2) + "\t" + rs.getString(3) + "\t" + rs.getString(4));

}

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[2].equals("sid"))

System.***out***.println("The SID is invalid");

**else** **if**(token[2].equals("student"))

System.***out***.println("The student has not taken any course");

//System.out.println ("\n\*\*\* SQLException caught \*\*\*\n" + token[2]);

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **Fifth.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** oracle.jdbc.OracleTypes;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class is for showing all the prerequisites of the courses

\*/

**public** **class** Fifth{

String deptcode;

**int** courseno;

/\*\*

\* **@param** dept\_code

\* **@param** course\_no

\*/

**public** Fifth(String dept\_code, String course\_no){

**this**.deptcode=dept\_code;

**this**.courseno=Integer.*parseInt*(course\_no);

}

/\*\*

\* **@throws** SQLException

\*/

**public** **void** execute\_fifth() **throws** SQLException{

Connection conn = **null**;

CallableStatement cs =**null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{ call prereq(?,?,?)}");

//set all the in parameters CHANGE IN HERE

cs.setString(1, deptcode); //dept\_code

cs.setInt(2, courseno); //course#

//register the out parameter (the first parameter)

cs.registerOutParameter(3, OracleTypes.***CURSOR***);

// execute and retrieve the result set

cs.executeUpdate();

ResultSet rs = (ResultSet)cs.getObject(3);

// print the results

**while** (rs.next()) {

System.***out***.println(rs.getString(1) + "\t" + rs.getInt(2));

}

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[0].equals("There"))

System.***out***.println("There are no prerequisites for the given course");

**else** **if**(token[0].equals("This"))

System.***out***.println("This course is not valid");

**else**

System.***out***.println("Invalid Entry");

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **sixth.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** oracle.jdbc.OracleTypes;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class is constructed to execute Query number 6 to show

\* all students in particular class

\*/

**public** **class** sixth {

String classid;

/\*\*

\* **@param** inclassid

\*/

**public** sixth(String inclassid){

**this**.classid = inclassid;

}

/\*\*

\* **@throws** SQLException

\* establish the connection and fetch records

\*/

**public** **void** show\_studinclass() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs =**null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{ call show\_studinclass(?,?)}");

//set all the in parameters

cs.setString(1, classid); //classid

//register the out parameter (the first parameter)

cs.registerOutParameter(2, OracleTypes.***CURSOR***); // the system\_refcursor to obtain the data

// execute and retrieve the result set

cs.executeUpdate();

ResultSet rs = (ResultSet)cs.getObject(2);

// print the results

**while** (rs.next()) {

System.***out***.println(rs.getString(1) + "\t" + rs.getString(2) + "\t" + rs.getString(3) );

}

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[2].equals("class"))

System.***out***.println("The CLASSID is invalid");

**else** **if**(token[2].equals("student"))

System.***out***.println("NO student is enrolled in class");

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

//close connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **seventh.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.SQLException;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class executes the seventh query

\*/

**public** **class** seventh {

String sid, classid;

// get parameters from driver code

**public** seventh(String insid, String inclassid){

**this**.sid=insid;

**this**.classid =inclassid;

}

/\*\*

\* **@throws** SQLException

\* connect to database and retrieve records

\*/

**public** **void** enroll\_student() **throws** SQLException{

Connection conn = **null**;

CallableStatement cs = **null**;

**try** {

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{call stud\_enroll(?,?) }");

//set all the in parameters

cs.setString(1, sid); //sid

cs.setString(2, classid); //classid

// execute and insert the records

cs.executeUpdate();

System.***out***.println("\nThe Student has successfully registered.");

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[2].equalsIgnoreCase("sid"))

System.***out***.println("The SID is invalid");

**else** **if**(token[2].equalsIgnoreCase("classid"))

System.***out***.println("The CLASSID is invalid");

**else** **if**(token[2].equalsIgnoreCase("class"))

System.***out***.println("The class is closed");

**else** **if**(token[2].equalsIgnoreCase("student"))

System.***out***.println("The student is already in this class");

**else** **if**(token[2].equalsIgnoreCase("Overloaded"))

System.***out***.println("You are Overloaded");

**else** **if**(token[2].equalsIgnoreCase("enroll"))

System.***out***.println("Cannot enroll in more than 4 classes in the same year and same semester");

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **Eighth.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.SQLException;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class executes the eighth query which helps the student

\* drop a certain class

\*/

**public** **class** Eighth {

String sid,classid;

/\*\*

\* **@param** insid

\* **@param** inclassid

\*/

**public** Eighth(String insid, String inclassid){

**this**.sid=insid;

**this**.classid=inclassid;

}

/\*\*

\* **@throws** SQLException

\*/

**public** **void** drop\_class() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs = **null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{call drop\_class(?,?) }"); //CHANGES IN HERE

//set all the in parameters CHANGE IN HERE

cs.setString(1, sid); //sid

cs.setString(2, classid); //classid

// execute and retrieve the result set

cs.executeUpdate();

System.***out***.println("The Student dropped the class successfully.");

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[1].equalsIgnoreCase("SID"))

System.***out***.println("The SID is invalid");

**else** **if**(token[1].equalsIgnoreCase("CLASSID"))

System.***out***.println("The CLASSID is invalid");

**else** **if**(token[1].equalsIgnoreCase("drop"))

System.***out***.println("The drop is not permitted because another class uses it as a prerequisite");

**else** **if**(token[1].equalsIgnoreCase("student"))

System.***out***.println("This student is not enrolled in this or any of the classes");

**else** **if**(token[1].equalsIgnoreCase("class"))

System.***out***.println("The class has now no students");

**else**

System.***out***.println("Some Connection Error! Please debug");

System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close the connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}

1. **Ninth\_del\_stud.java**

**package** jdbc\_trial;

**import** java.sql.CallableStatement;

**import** java.sql.Connection;

**import** java.sql.SQLException;

**import** oracle.jdbc.pool.OracleDataSource;

/\*\*

\* **@author** Bhavin

\* This class is used to delete a student from the class.

\* Throw appropriate exceptions

\*/

**public** **class** Ninth\_del\_stud {

String sid;

/\*\*

\* **@param** insid

\*/

**public** Ninth\_del\_stud(String insid){

**this**.sid=insid;

}

/\*\*

\* **@throws** SQLException

\*/

**public** **void** delete\_student() **throws** SQLException {

Connection conn = **null**;

CallableStatement cs =**null**;

**try**

{

//Connection to Oracle server

OracleDataSource ds = **new** oracle.jdbc.pool.OracleDataSource();

ds.setURL("jdbc:oracle:thin:@grouchoIII.cc.binghamton.edu:1521:ACAD111");

conn = ds.getConnection("bdesai1", "Quantum900");

//Prepare to call stored procedure:

cs = conn.prepareCall("{call del\_stud(?) }");

//set all the in parameters

cs.setString(1, sid); //sid

// execute and retrieve the result set

cs.executeUpdate();

System.***out***.println("Records successfully deleted");

//close the result set, statement, and the connection

cs.close();

conn.close();

}

**catch** (SQLException ex) {

// get the error messages based on requirements

String message = ex.getMessage();

String token[] = message.split(" ");

**if**(token[2].equalsIgnoreCase("sid"))

System.***out***.println("The SID is invalid");

}

**catch** (Exception e) {System.***out***.println ("\n\*\*\* other Exception caught \*\*\*\n");}

**finally** {

// close connection

**if**(cs!=**null** || conn!=**null**){

**try**{

cs.close();

conn.close();

}

**catch**(SQLException ex){System.***out***.println ("\n\*\*\* SQLException caught \*\*\*\n" + ex.getMessage());}

}

}

}

}